

CS 20, Spring 2002

Introduction to Scientific Programming

Daniel Kaplan, Macalester College

The Poincaré Oscillator and Data Analysis

The purpose of this exercise is to explore some aspects of the link between nonlinear dynamics and data analysis. We will generate data from a simple but important nonlinear system, the Poincaré oscillator, and analyze these data to extract information about the system.

The forced Poincaré oscillator is a model of the sleep-wake system. In radial coordinates the unforced oscillator is given by a pair of uncoupled differential equations:

$$\frac{dr}{dt} = \alpha r(1 - r) \quad (1)$$

$$\frac{d\theta}{dt} = \beta \quad (2)$$

The dynamics of the unforced oscillator are simple: the state cycles around the unit circle, the cycle length is set by the parameter β and the speed of approach to the unit circle (that is, the points where $r = 1$) is set by α .

The differential equations could also be written in rectangular coordinates, with $x = r \cos \theta$ and $y = r \sin \theta$. If the state is visualized as a point in a plane, then the type of forcing we will study involves an input that moves the state point over to the right. That is, the forced oscillator is described by

$$\frac{dx}{dt} = f_x(x, y) + u(t) \quad (3)$$

$$\frac{dy}{dt} = f_y(x, y) \quad (4)$$

where $u(t)$ is the input and f_x and f_y are the functions implicit in Eqs 1 and 2.

In experimental studies we do not have access to the state variables (r, θ) (or, equivalently, (x, y)). Instead, we measure a quantity that presumably depends on the state. In this exercise, we will measure a Boolean (0 or 1) variable that indicates wakefulness. Thus, our measured time series will look like a square wave.

We will apply inputs $u(t)$ and observe the output $s(t)$. The objective is to use the $u(t) \rightarrow s(t)$ pairs to make deductions about the system. This should, in principle, be easy, since we know the dynamics of the system. But there are substantial difficulties. First, standard forms of $u(t)$ produce steady state behavior. Near the steady state, important aspects of the dynamics are not evident in the dynamics. Second, the $s(t)$ measurement is not in a form that is directly useful for inferring the dynamics.

The primary software for this exercise is the m-file `sleepcycle` which provides an Euler integration of Eqs 3 and 4. (There is no good mathematical reason to use an Euler integration. In some circumstances, the dynamics of Eqs 3 and 4 and the program `sleepcycle` are quite different. Perhaps it's best to think of Eq. 3 as a model of how the data are generated, rather than being reality.)

`Sleepcycle` is written with a time step of 5 minutes, so that there are 288 points in a 24-hour day. The input $u(t)$ should be arranged as a vector with one point for each 5-minute time step. The input is intended to be interpreted as light: 0 means darkness and 1 is a "usual" level of light. Typically, in a Poincaré model of sleep-wake cycle, the duration of the input is much shorter than the length of daylight, perhaps corresponding to an accommodation of the light sensitivity mechanism as the day wears on.

Here is an input that corresponds to 100 days of a 24-hour periodic exposure to light, where the effective duration of the light each day is approximately 4 hours (50 out of 288 points):

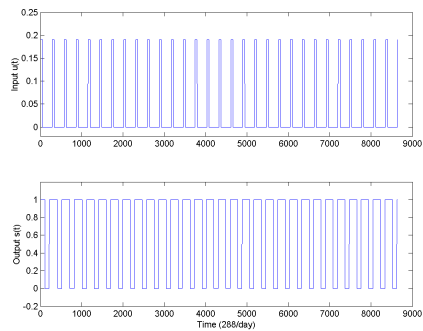


Figure 1: A periodic input and the corresponding output.

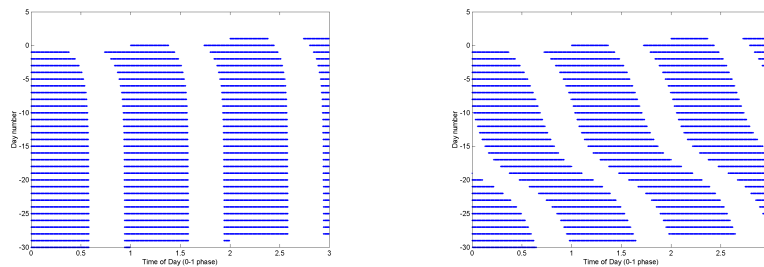


Figure 2: Left: the rasterplot corresponding to the output signal in the above figure. There is phase locking. Right: a signal of smaller amplitude (0.15) fails to produce phase locking.

```
>> u = rem(1:(100*288), 288) < 50;
```

To compute the output, one runs `sleepcycle` with `u` as an input:

```
>> s = sleepcycle(u);
```

In this case, both $u(t)$ and $s(t)$ are square waves.

The program `rasterplot` takes $s(t)$ as input and plots it in the manner used for sleep-wake studies. This format of plot is useful, but is by no means the only way of analyzing the data. For example, the times of transition between sleep and wakefulness can be computed with

```
>> wakeuptimes = find(diff(s) < 0);
```

```
>> gotosleeptimes = find(diff(s) > 0);
```

(These times will be in terms of the sampling time of 5 minutes.) Even when the system is in steady state, the timing shift between the onset of light and the onset of sleep or wake provide one indication of the dynamics.

Your task is to provide as complete a description as possible of the sleep-wake dynamics. You can select inputs that you think are particularly illuminating, perhaps varying the period and amplitude. You may also want to construct experiments in which you draw out the phase-response curve.